

924



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/852,070	05/09/2001	Daniel K. Hiltgen	SMQ-122/P6281	3533
959	7590	07/01/2004	EXAMINER	
LAHIVE & COCKFIELD, LLP. 28 STATE STREET BOSTON, MA 02109			YIGDALL, MICHAEL J	
			ART UNIT	PAPER NUMBER
			2122	

DATE MAILED: 07/01/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

## Office Action Summary

Application No.

09/852,070

Applicant(s)

HILTGEN ET AL.

Examiner

Michael J. Yigdall

Art Unit

2122

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 09 May 2001.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-32 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-32 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 09 May 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date 5/12/2002
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

### **DETAILED ACTION**

1. Claims 1-32 are pending and have been examined. The priority date considered for the application is May 9, 2001.

#### ***Specification***

2. The disclosure is objected to because of the following informalities: The phrase "many-to-may" on page 5, line 23 (paragraph 0014) was perhaps intended to be --many-to-many--. As well, there are numerous inconsistencies on page 8 (paragraph 0019) with regard to the drawings. For example, the reference to the "verified flag 56" in FIG. 5 should perhaps refer instead to element 74 in FIG. 4 (reference number 56 does not appear in the drawings). Likewise, "realization entry 60," "realization list 52," "realization name 50" and "version number 42" conflict with what is shown in the drawings. On page 12 (paragraph 0026), references are made to both the "patch realization mapping 28" and the "patch expression 28." Furthermore, the reference to FIG. 10 on page 14, line 5 (paragraph 0032) should perhaps refer instead to FIG. 9, and in line 14, the reference to an absent FIG. 11 should perhaps refer instead to FIG. 10.

Applicant is respectfully asked for cooperation in finding all such errors and inconsistencies in the disclosure. Appropriate correction is required.

#### ***Double Patenting***

3. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed.

Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970);and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

4. Claims 1-32 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-58 of copending Application No. 09/851,923. Although the conflicting claims are not identical, they are not patentably distinct from each other because both recite analogous methods, systems and articles of manufacture for determining whether to install a patch on a computer system.

For example, claims 1 and 7 of the instant application and claim 1 of App. No. 09/851,923 both recite determining configuration information regarding the hardware and software components of a computer system and defining realization states associated with the patches that may be applied. The conditional statements and patch attribute statements recited by instant claim 1 are similar to the patch expressions and realization detectors of claims 7 and 8 in App. No. 09/851,923 for determining whether the patch should be installed. Likewise, the list of patches recited by instant claim 1 is also recited by claim 10 of App. No. 09/851,923.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

5. Claims 1-32 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-52 of copending Application No. 09/852,113. Although the conflicting claims are not identical, they are not patentably distinct from each other because both recite analogous methods, systems and articles of manufacture for determining whether to install a patch on a computer system.

For example, claims 1 and 7 of the instant application and claim 1 of App. No. 09/852,113 both recite a list of patches and associated realization states for determining whether to install a patch on a computer system. Claim 7 of App. No. 09/852,113 recites patch expression sets that are analogous to the conditional statements recited by instant claim 1. The computer object recited in instant claim 1 is likewise recited by claim 8 of App. No. 09/852,113; both claims recite processing the configuration information, which relates to the components of the computer system, to determine whether to install the patch.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

#### ***Claim Rejections - 35 USC § 103***

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 1, 5, 7-9, 13, 15-17, 21, 23-25, 29, 31 and 32 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,751,794 to McCaleb et al. (hereinafter "McCaleb") in view of U.S. Pat. No. 6,279,030 to Britton et al. (hereinafter "Britton").

With respect to claim 1, McCaleb discloses a method for creating a patch including content to apply to a computer (see the title and abstract), comprising:

(a) generating a script program including at least one patch attribute statement, wherein each patch attribute statement is called with at least one conditional statement that returns a list of one or more patches if the at least one conditional statement evaluates as true based on a presence of a software or hardware component indicated in a computer object for the computer on which the patch will be applied, and wherein an attribute defined for the attribute statement is associated with the installation of the patch to the computer if the computer includes the returned list of patches (see column 4, line 59 to column 5, line 23, which shows generating a script with parameters or attributes for checking the software configuration of the client computer and returning a corresponding list of applications that may be updated); and

(b) associating the script program with the patch, wherein the script program executes and processes the computer object including information on installed software and hardware components to determine whether to install the patch on the computer based on attributes of the installation determined by the script program (see column 5, lines 24-50, which shows executing the script and processing the information from the client computer to determine whether to install an update patch).

Although McCaleb does not expressly disclose the limitation wherein the conditional statements evaluate to true or false, Britton discloses conditional predicates or statements with attributes that evaluate to true or false based on Boolean logic principles, the results of which are used to select a version of a component (see column 9, line 48 to column 10, line 13).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the patch checking system of McCaleb with the conditional statements taught by Britton, in order to facilitate determining whether to install a version of a patch.

With respect to claim 5, McCaleb in view of Britton further discloses the limitation wherein the patch attribute statements are further capable of including a patch constraint attribute called with at least one conditional statement, wherein the patch constraint attribute indicates that the patch can be installed in the computer if each conditional statement evaluates as true (see Britton, column 9, line 48 to column 10, line 13, which shows conditional predicates or statements with attributes or constraints that, when they evaluate to true, indicate that a component or patch may be selected).

With respect to claim 7, McCaleb in view of Britton further discloses the limitation wherein the computer object is further capable of indicating a realization that defines a state of the computer, wherein the conditional statements are further capable of determining whether realizations are included in the computer object indicating the presence of the defined realization states at the computer, wherein the attribute for the patch attribute statement is associated with the installation of the patch to the computer if the at least one conditional statement is evaluated as true (see McCaleb, column 6, lines 34-39, which shows information indicating the hardware

Art Unit: 2122

and software state of the client computer, and column 5, lines 24-50, which shows such information used in association with an update or patch to be installed; see also Britton, column 9, line 48 to column 10, line 13, which shows selecting a component when the corresponding conditional predicates or statements evaluate to true).

With respect to claim 8, McCaleb in view of Britton further discloses the limitation wherein the patch attribute statement includes multiple conditional statements and a different list of patches for each conditional statement, wherein the attribute defined for the attribute statement is associated with the installation of the patch to the system if the system includes the returned list of patches for the conditional statement that evaluated as true (see McCaleb, column 5, lines 51-54, which shows that there may be several files to update for each software component that is checked, i.e. a different list of patches for each conditional statement; see also Britton column 9, line 48 to column 10, line 13, which shows selecting a component when the corresponding conditional predicates or statements evaluate to true).

With respect to claim 9, McCaleb discloses a system for determining whether to apply a patch including content onto a computer (see the title and abstract), comprising:

- (a) a processor (see column 7, lines 59-67);
- (b) a computer readable medium including a computer object including information on installed software and hardware components determined in the computer (see column 6, lines 34-39, which shows a database including information on installed software and hardware components of a client computer);



(c) an interpreter program capable of interpreting (see column 5, lines 24-50, which shows a patch checker for interpreting information from the client computer to determine whether to install an update patch):

(i) a set of conditional statements that return a Boolean response based on a presence of a software or hardware component indicated in the computer object for the computer on which the patch will be applied (see column 4, line 59 to column 5, line 23, which shows conditional statements for checking the software configuration of the client computer on which the patch will be applied);

(ii) a patch attribute statement called with at least one conditional statement that returns a list of one or more patches if the at least one conditional statement evaluates as true, wherein a attribute defined for the attribute statement is associated with the installation of the patch to the computer if the computer includes the returned list of patches (see column 4, line 59 to column 5, line 23, which shows parameters or attributes for checking the client computer and returning a corresponding list of applications that may be updated); and

(iii) a script program including at least one patch attribute statement, wherein the script program statements and computer object are processed to determine whether the at least one conditional statement associated with each patch attribute statement is true and, if so, whether the patches in the returned list are included in the computer (see column 4, line 59 to column 5, line 23, which shows a script with parameters or attributes for checking the software configuration of the client computer).

Although McCaleb does not expressly disclose the limitation wherein the conditional statements return a Boolean response and evaluate to true or false, Britton discloses conditional predicates or statements with attributes that evaluate to true or false based on Boolean logic principles, the results of which are used to select a version of a component (see column 9, line 48 to column 10, line 13).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the patch checking system of McCaleb with the conditional statements taught by Britton, in order to facilitate determining whether to install a version of a patch.

With respect to claims 13, 15 and 16, the limitations recited by these claims are analogous to the limitations recited by claims 5, 7 and 8, respectively. Therefore, the explanations for claims 5, 7 and 8 provided above apply to claims 13, 15 and 16 as well, respectively.

With respect to claim 17, McCaleb discloses an article of manufacture including program code for determining whether to apply a patch including content to a computer (see the title and abstract, and column 7, lines 47-58) by:

(a) providing a computer object including information on software and hardware components in the computer (see column 6, lines 34-39, which shows a database including information on installed software and hardware components of a client computer);

(b) interpreting a script program including at least one patch attribute statement, wherein each patch attribute statement is called with at least one conditional statement that returns a list of one or more patches if the at least one conditional statement evaluates as true based on a

presence of a software or hardware component indicated in the computer object for the computer on which the patch will be applied and wherein an attribute defined for the attribute statement is associated with the installation of the patch to the computer if the computer includes the returned list of patches (see column 4, line 59 to column 5, line 23, which shows a script with parameters or attributes for checking the software configuration of the client computer and returning a corresponding list of applications that may be updated); and

(c) processing the computer object including information on installed software and hardware components when executing the script program to determine whether to install the patch on the computer based on attributes of the installation determined by the script program (see column 5, lines 24-50, which shows processing the information from the client computer to determine whether to install an update patch).

Although McCaleb does not expressly disclose the limitation wherein the conditional statements evaluate to true or false, Britton discloses conditional predicates or statements with attributes that evaluate to true or false based on Boolean logic principles, the results of which are used to select a version of a component (see column 9, line 48 to column 10, line 13).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the patch checking system of McCaleb with the conditional statements taught by Britton, in order to facilitate determining whether to install a version of a patch.

With respect to claims 21, 23 and 24, the limitations recited by these claims are analogous to the limitations recited by claims 5, 7 and 8, respectively. Therefore, the explanations for claims 5, 7 and 8 provided above apply to claims 21, 23 and 24 as well, respectively.

With respect to claim 25, McCaleb discloses a computer readable medium including data structures for determining whether to apply a patch including content to a computer (see the title and abstract, and column 7, lines 47-58), comprising:

(a) a script program including at least one patch attribute statement wherein each patch attribute statement is called with at least one conditional statement that returns a list of one or more patches if the at least one conditional statement evaluates as true based on a presence of a software or hardware components in the computer on which the patch will be applied, and wherein an attribute defined for the attribute statement is associated with the installation of the patch to the computer if the computer includes the returned list of patches, and wherein the script program and information on installed software and hardware components are processed to determine whether to install the patch on the computer based on attributes of the installation determined by the script program (see column 4, line 59 to column 5, line 23, which shows a script with parameters or attributes for checking the software configuration of the client computer and returning a corresponding list of applications that may be updated, and column 5, lines 24-50, which shows processing the information from the client computer to determine whether to install an update patch).

Although McCaleb does not expressly disclose the limitation wherein the conditional statements evaluate to true or false, Britton discloses conditional predicates or statements with attributes that evaluate to true or false based on Boolean logic principles, the results of which are used to select a version of a component (see column 9, line 48 to column 10, line 13).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the patch checking system of McCaleb with the conditional statements taught by Britton, in order to facilitate determining whether to install a version of a patch.

With respect to claims 29, 31 and 32, the limitations recited by these claims are analogous to the limitations recited by claims 5, 7 and 8, respectively. Therefore, the explanations for claims 5, 7 and 8 provided above apply to claims 29, 31 and 32 as well, respectively.

8. Claims 2, 10, 18 and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over McCaleb in view of Britton as applied to claims 1, 9, 17 and 25 above, respectively, and further in view of U.S. Pat. No. 6,442,754 to Curtis.

With respect to claim 2, although McCaleb discloses that each update may comprise more than one file and that more than one update may be needed (see column 5, lines 51-54), McCaleb does not expressly disclose the limitation wherein the patch attribute statement is capable of comprising a patch requires statement, wherein the patch requires attribute indicates that the patches in the returned patch list must be installed in the computer in order for the patch to be installed in the computer.

However, Curtis discloses a dependency object for a program to be installed that indicates whether other files or programs are required and must also be installed (see column 9, lines 21-31).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the patch checking system of McCaleb with a patch requires statement,

such as the dependency object taught by Curtis, in order to indicate whether other patches must also be installed in the computer.

With respect to claims 10, 18 and 26, the limitations recited by these claims are analogous to the limitations recited by claim 2. Therefore, the explanation for claim 2 provided above applies to claims 10, 18 and 26 as well.

9. Claims 3, 11, 19 and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over McCaleb in view of Britton as applied to claims 1, 9, 17 and 25 above, respectively, and further in view of U.S. Pat. No. 6,266,811 to Nabahi.

With respect to claim 3, although McCaleb discloses attribute statements that are similar to commands interpreted by the operating system (see column 5, lines 1-6), McCaleb does not expressly disclose the limitation wherein the conditional and patch attribute statements utilize a syntax that is similar to a syntax of commands capable of being interpreted by a command processor interface of the computer operating system, and wherein the syntax of the conditional and patch attribute statements prevent the conditional and patch attribute statements from executing on the computer outside of a patch update interpreter that is capable of interpreting the syntax of the conditional and patch attribute statements.

However, Nabahi discloses a scripting language for installing software packages comprised of commands that are compiled and executed solely by a rule-based installation engine, i.e. a patch update interpreter (see column 5, lines 22-42), with a syntax that is similar to statements associated with the command processor interface of the operating system (see column 13, line 11 to column 14, line 64, which shows the syntax of some such commands).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to implement the statements of McCaleb using a command syntax that is similar to one understood by the operating system and is interpreted solely by the installation engine or update interpreter, as taught by Nabahi, so that the script would be intuitive (see Nabahi, column 14, line 65 to column 15, line 5) and compiled to enhance security (see Nabahi, column 2, line 57 to column 3, line 2).

With respect to claims 11, 19 and 27, the limitations recited by these claims are analogous to the limitations recited by claim 3. Therefore, the explanation for claim 3 provided above applies to claims 11, 19 and 27 as well.

10. Claims 4, 12, 20 and 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over McCaleb in view of Britton as applied to claims 1, 9, 17 and 25 above, respectively, further in view of Curtis and further in view of U.S. Pat. No. 6,513,159 to Dodson.

With respect to claim 4, although McCaleb discloses checking the software configuration of the client computer to determine the versions of several applications (see column 5, lines 11-24), McCaleb does not expressly disclose the limitation wherein the patch attribute statements included in the script program are capable of including patch attribute statements that are members of the set of patch attribute statements comprising:

(a) a patch incompatible statement wherein the patch incompatible attribute indicates that if the patches in the returned patch list are installed in the computer, then the patch cannot be installed in the computer; and

(b) a patch prefers statement, wherein the patch prefers attribute indicates that the patches in the returned patch list are recommended to be installed in the computer.

However, Dodson discloses a list of packages known to work properly together that is used to identify incompatible combinations of drivers and devices when installing or updating another package (see column 3, lines 21-49).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the patch checking system of McCaleb with a patch incompatible statement, in order to indicate whether the patch is incompatible with other patches installed in the computer, such as is achieved by the package list taught by Dodson.

Furthermore, Curtis discloses a dependency object for a program to be installed that indicates whether other files or programs are required and must also be installed (see column 9, lines 21-31). The dependency object may also be set such that it does not actually check for the other files and programs, and thus serves instead as list of preferred or recommended files (see column 9, lines 47-50).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the patch checking system of McCaleb with a patch prefers statement, such as the dependency object taught by Curtis, in order to indicate whether other patches are recommended to be installed in the computer.

With respect to claims 12, 20 and 28, the limitations recited by these claims are analogous to the limitations recited by claim 4. Therefore, the explanation for claim 4 provided above applies to claims 12, 20 and 28 as well.



11. Claims 6, 14, 22 and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over McCaleb in view of Britton as applied to claims 1, 9, 17 and 25 above, respectively, and further in view of Dodson.

With respect to claim 6, although McCaleb discloses information indicating the hardware and software state of the client computer (see column 6, lines 34-39), McCaleb does not expressly disclose the limitation wherein the conditional statements provided with the attribute statements are members of the set of conditional statements comprising:

(a) a first conditional statement that determines whether the computer object indicates that a specified vendor operating system is installed on the computer;

(b) a second conditional statement that determines whether the computer object indicates that a specified version of the operating system is installed on the computer;

(c) a third conditional statement that determines whether the computer object indicates that the computer includes a specified hardware platform; and

(d) an eighth conditional statement that determines whether the computer object indicates that the computer includes a specified architecture.

However, Dodson discloses checking the type and version of the operating system installed on the computer, the hardware configuration of the computer, and the architecture of the computer, in order to determine which drivers to install or update (see column 4, line 59 to column 5, line 43).

McCaleb further discloses the limitation wherein the conditional statements provided with the attribute statements are members of the set of conditional statements comprising:

(e) a fourth conditional statement that determines whether the computer object indicates that a specified software package having a specified version number is installed on the computer (see column 5, lines 24-50, which shows determining whether a software package installed on the client computer has a specific version number);

(f) a fifth conditional statement that determines whether the computer object indicates that a specified software package having a specified version number or higher is installed on the computer (see column 5, lines 24-50, which shows determining whether a software package installed on the client computer is up to date, i.e. has a specific version number or higher);

(g) a sixth conditional statement that determines whether the computer object indicates that a specified patch having a specified version number is installed on the computer (see column 5, lines 24-50, which shows determining whether a software package installed on the client computer, including any corresponding patches, has a specific version number); and

(h) a seventh conditional statement that determines whether the computer object indicates that a specified patch package having a specified version number or higher is installed on the computer (see column 5, lines 24-50, which shows determining whether a software package installed on the client computer, including any corresponding patches, is up to date, i.e. has a specific version number or higher).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the conditions taught by Dodson with the conditional statements of McCaleb, for the purpose of determining the operating system, hardware platform and architecture of the computer when checking a patch to be installed.

With respect to claims 14, 22 and 30, the limitations recited by these claims are analogous to the limitations recited by claim 6. Therefore, the explanation for claim 6 provided above applies to claims 14, 22 and 30 as well.

### ***Conclusion***

12. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. U.S. Pat. No. 6,199,204 to Donohue discloses a method for distributing software updates. U.S. Pat. No. 6,282,711 to Halpern et al. discloses a method for installing software components from a server. U.S. Pat. No. 6,367,073 to Elledge discloses a method for installing software. U.S. Pat. No. 6,735,766 to Chamberlain et al. discloses a method for installing an upgrade to an application.

13. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Yigdall whose telephone number is (703) 305-0352. The examiner can normally be reached on Monday through Friday from 7:30am to 4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (703) 305-4552. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

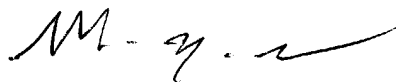
Art Unit: 2122

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

MY

Michael J. Yigdall  
Examiner  
Art Unit 2122

mjy



**WEI Y. ZHEN**  
**PRIMARY PATENT EXAMINER**